2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Workshops

Efficient Allocation of Hierarchically-Decomposable Tasks in a Sensor Web Contract Net

John S. Kinnebrew and Gautam Biswas
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN 37203, USA
Email: john.s.kinnebrew@vanderbilt.edu

***Abstract*—In large, distributed systems, such as a sensor web, allocating resources to tasks that span multiple providers presents significant challenges. Individual subtasks associated with a task could potentially be assigned to a number of agents (*e.g.*, when there is overlap in sensor or data processing capability among constituent sensor networks). This problem is further compounded by the dynamic nature of a sensor web, in which both desired tasks and resource availability change with time and environmental conditions. This paper presents a novel variation of the contract net protocol (CNP) for subtask allocation, which employs brokers to limit communication overhead in a two-phase CNP and aggregate domain information from groups of agents. Experimental results using this subtask allocation approach verify its efficiency and scalability. These results also suggest specific refinements and appropriate parameters for a variety of system configurations and operating conditions in sensor webs and other large multi-agent systems.**

I. Introduction

The availability of a variety of sensors around the globe and the ability to quickly make this data available in remote locations give today's scientists an unprecedented advantage in studying and predicting weather, natural disasters, and climate change. Selecting and coordinating an appropriate subset of these heterogeneous and distributed sensors for such large-scale tasks is complex. For example, sensors must first be located and may have to be reconfigured and recalibrated to collect the needed data. Moreover, many complex tasks require the cooperation of multiple component sensor networks. Coordination among these distributed sensor and computational resources is required for efficient and effective execution of complex tasks.

As illustrated in Figure 1, a sensor web is made up of many independent sensor networks. One difficulty in task allocation for a sensor web is that available resources (*e.g.*, sensors, servers, bandwidth) are not owned or controlled by any single entity. Various institutions, governments, and corporations will have the final say on how their resources are deployed and used. Further, a global sensor web will have many independent, heterogeneous "users" (*e.g.*, weather modeling and prediction systems, disaster recognition and management systems, and scientists) requesting access to, and control of, the sensor platforms to support their research and analysis activities. A multi-agent system (MAS) provides a natural approach for distributed, resource-aware coordination and control in a large-scale system composed of heterogeneous, independent entities.

Sensor web users may often request allocation of tasks requiring resources from multiple independent sensor networks, each represented by an independent agent. Further, when the overall task is broken down into subtasks, an individual subtask may require resources that could be provided by multiple agents (*e.g.*, when there is overlap in sensor or data processing capability between multiple sensor networks in the sensor web). Consequently, there may be many combinations of agents capable of executing the overall task, as represented by the possible subtask allocations, which will be of varying utility to the requesting agent. This problem requires an efficient mechanism for achieving a high utility allocation of subtasks among applicable agents.

This paper presents an empirical study of the performance of a novel variation of the contract net protocol for allocating hierarchically-decomposable tasks in a large multi-agent system. This extension of the contract net protocol employs brokers to efficiently limit communication overhead while generating high-utility allocations of subtasks in a two-phase contract net. First, Section II provides an overview of the motivating framework for this work and identifies the challenges in allocating complex tasks in a sensor web. Next, we outline our approach to efficiently generating a high-utility subtask allocation with a brokered, two-phase contract net in Section III. Section IV presents the results of our experiments in contract net subtask allocation under a variety of conditions and analyzes those results to verify the scalability and effectiveness of this approach. We discuss the implications of these results for real-world multi-agent systems and provide a comparison to related work in Section V. Finally, Section VI presents concluding remarks and suggests specific extensions for future work.

II. The Multi-agent Architecture for Coordinated Responsive Observations

This paper addresses the problem of allocation for hierarchically-decomposable tasks in the context of large-

978-0-7695-3801-3/09 $26.00 © 2009 IEEE
DOI 10.1109/WI-IAT.2009.154

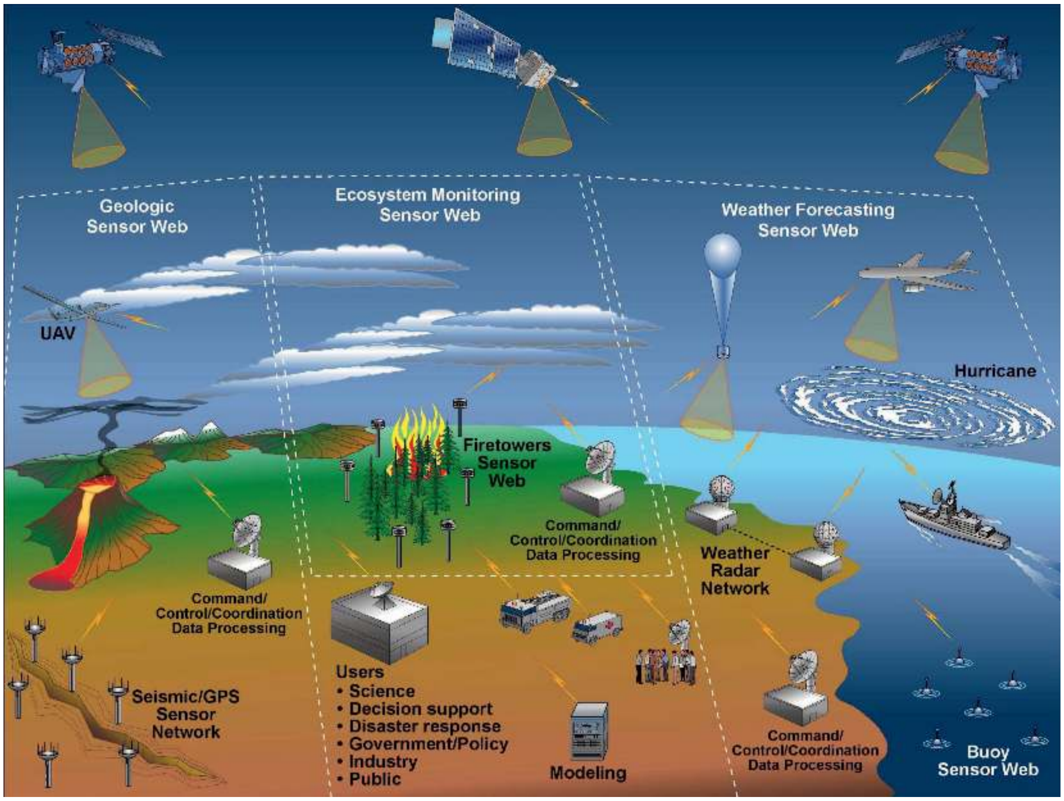

Figure 1. A Global Sensor Web [1]

scale multi-agent systems, in general, and the *Multi-agent Architecture for Coordinated Responsive Observations* (MACRO) platform [2], in particular. MACRO provides a powerful computational infrastructure for enabling the deployment, configuration, and operation of large sensor webs that are composed of many constituent sensor networks. MACRO is divided into a broad, two-level hierarchy of agents: (1) the *mission level*, where agents interact with users to allocate high-level science tasks to sensor net resources and coordinate to create plans and schedules to achieve these tasks, and (2) the *resource level*, where local server and field agents translate tasks into actions and application deployments related to configuration and data collection, analysis, and transmission [2]. In this paper, we focus on the mission level of MACRO, where user tasks are allocated to agents with the resources to achieve them.

The MACRO mission level is comprised of *user agents*, *mission agents*, and *broker agents*. User agents are the primary providers of high-level tasks to be achieved by the system. Typically the user agents are interfaces to mission scientists or wrappers for legacy systems (*e.g.* weather modeling applications) that can request execution of sensor web tasks. Each mission agent represents an independent *sensor network* and achieves its allocated tasks with the resources available in its sensor network. Finally, broker agents act as an intelligent system infrastructure at the mission level, providing matchmaker services, aggregating relevant domain information, tracking system performance, and mediating allocation negotiations [3].

The complex nature of tasks and plans at the mission level of a sensor web MAS makes hierarchical analysis important for dealing with this complexity, both for problem/task representation by domain experts and for coordinated planning among multiple agents. MACRO employs a modified implementation of the Task Analysis, Environment Modeling, and Simulation (TÆMS) [4] language, which provides a hierarchical task tree representation for multi-agent planning and scheduling. Further, MACRO incorporates the OGC SensorML [5] representation of sensors and data processing with the TÆMS hierarchically decomposable task representation to provide standardized descriptions of task/subtask requirements and effects.

Employing a hierarchical decomposition representation of tasks, such as TÆMS, in a sensor web MAS presents a significant challenge. TÆMS and similar hierarchical decomposition structures were primarily designed for systems built by a single group of designers sharing a conception of the entire system and its function. In that context, task decomposition trees can be built with a top-down goal-oriented structure combined with bottom-up knowledge of the functionality available in the system. However, at the mission level in MACRO, mission agents and their domain knowledge, including TÆMS task trees, are implemented by a variety of groups, such as individual organizations operating sensor networks. The tasks they include are specific to their individual purposes and capabilities without full knowledge of related or equivalent tasks throughout the system. This presents a challenge in combining mission agent task trees and determining potential decompositions of requested tasks that require resources of multiple mission agents. MACRO's broker-based solution to this challenge is discussed in Section III.

In MACRO, sensor web tasks are allocated using a modified Contract Net Protocol (CNP) [6] in order to accommodate both fairness and system utility considerations [3]. The CNP is a widely implemented MAS solution to task/resource allocation that uses computationally simple, single auctions. In particular, the CNP and its derivatives can allow effective allocation of tasks/resources without restricting the criteria individuals agents can apply to determine their preference for tasks or bids. This allows significant flexibility in agent design and is especially important when individual agents are designed by different parties with varying internal goals and constraints, as in a sensor web.

The first step in allocating high-level tasks is to determine which agents are capable of executing part or all of the task. In the MACRO contract net, broker agents are employed to provide this matchmaker service. This significantly reduces the communication required to allocate tasks compared to the traditional CNP, in which each task announcement is sent to every agent. This brokered contract net provides a negotiation framework for task allocation in which user agents contract with relevant mission agents to achieve tasks.

A major challenge in applying the contract net protocol to a sensor web MAS is its emphasis on two-party contracts. In MACRO, a user agent may require the resources of multiple mission agents to achieve its high-level task. Therefore, MACRO uses subcontracting, allowing resources of multiple mission agents to be assigned to a task through a primary contract and additional subcontracts between mission agents. However, with multiple possible decompositions of a task, there may be a large number of mission agents who could be the primary contractor for the task, and each mission agent could choose from a variety of subcontractors. If each of these mission agents announces subtasks and receives bids before bidding on the task, a great deal of communication and computation must be performed before each can generate a complete bid on the task. On the other hand, if each mission agent bids on the task before soliciting subcontracts, there is a great deal of uncertainty in the accuracy of resulting bids. Finding a trade-off between subcontracting overhead and completeness of bids for decomposable tasks presents a major challenge for efficient and effective allocation. The MACRO solution approach to this challenge is detailed in Section III and verified by experimental studies in Section IV.

III. MACRO Approach to Subtask Allocation

MACRO extends the traditional contract net protocol with broker agents and limited, two-phase contracting to efficiently allocate hierarchically-decomposable tasks. MACRO employs two types of broker agents to mediate the contract net negotiations, as illustrated in Figure 2. Although a single type of broker agent is theoretically sufficient to perform all necessary services, MACRO divides broker agents into two tiers based on their specific roles and responsibilities to simplify system deployment and dynamic modification [3].

One responsibility of MACRO brokers is to provide an efficient matchmaking/locater service (*i.e.*, determining agents capable of performing all or part of an announced task and forwarding messages appropriately). Tier 2 broker agents cluster mission agents by geographic region and maintain a directory of sensor and computational capabilities for the mission agents in their region. The requirements of an announced task are used by tier 1 broker agents to forward task announcements to appropriate tier 2 broker agents, who then relay the announcement to applicable mission agents. Tier 1 broker agents are also responsible for assigning system utility values to task announcements taking into account user agent priority and task importance, as well as past system performance. Even though user agents and mission agents may be implemented by parties other than the system designers, broker valuation of tasks allows the system to influence the contract net negotiations, and optimize task allocation for appropriate efficiency and fairness metrics.

Another responsibility of th MACRO brokers is to aggregate domain knowledge across mission agents. Tier 2 brokers aggregate partial task trees from their assigned mission agents to determine how complex tasks spanning multiple sensor networks can be decomposed into subtasks, which can each be performed by individual mission agents. Since mission agent task trees may be designed independently, they are enhanced with standardized sensor web information (OGC SensorML [5]) regarding task/subtask requirements and effects. This allows MACRO brokers to determine appropriate join points and overlap between independent task trees. Tier 1 brokers further aggregate the partial task trees from Tier 2 brokers, ultimately providing conversion of high-level user tasks (initially expressed in an annotated SensorML format) to potential subtask decompositions. This allows MACRO to resolve the challenge, identified in Section II, of integrating independently-designed task trees across domains.

Section II also identified the challenge of balancing message overhead and bid completeness for efficient, effective subtask allocation. The MACRO CNP addresses this challenge by separating initial bidding from final bidding,

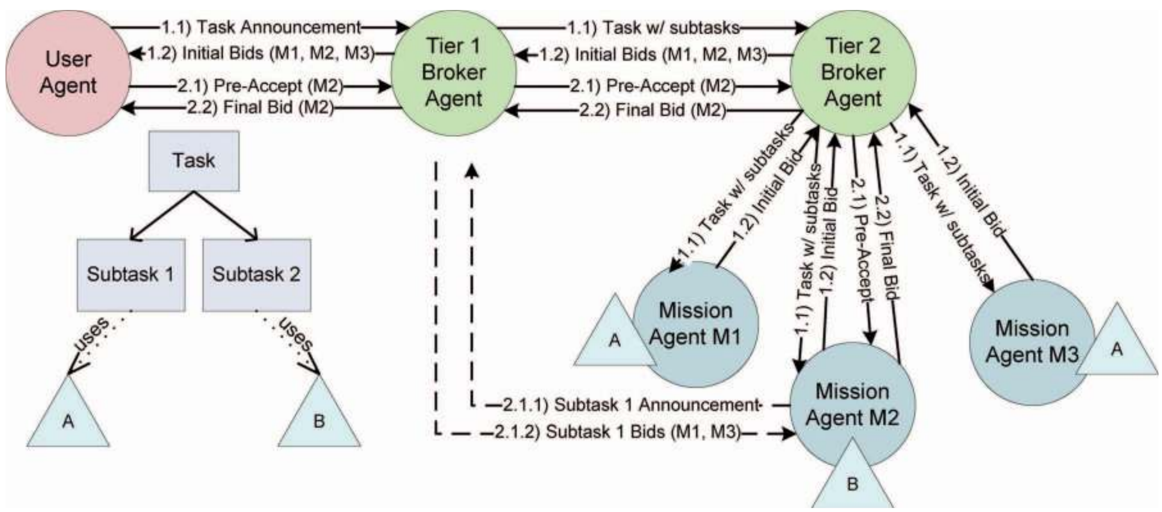

Figure 2. MACRO Contract Net

extending the approach in the two-phase contract net protocol proposed by Aknine et al. [7]. This variation of the CNP breaks the contract net negotiations into an initial pre-commitment phase and a final commitment phase. During pre-commitment, the task is announced and initial bids are made, as illustrated by the 1.x steps in Figure 2. The announcing agent then pre-accepts what it determines to be the best initial bid, beginning the commitment phase, illustrated by the 2.x steps. In the MACRO CNP the pre-accepted agent can then announce subtasks it may not be able to, or want to, perform. As illustrated in step 2.1.1, a pre-accepted Mission agent is acting in an announcing capacity for these subtasks and communicates them to an assigned Tier 1 Broker, just as the User agent did for the initial task announcement. After receiving bids on its announced subtasks, the Mission agent makes a final bid on the task. This final bid includes relevant information on subcontracts and better estimates of applicable quality measures and time-to-completion. The announcing agent can then accept the final bid or pre-accept a different agent.

MACRO employs this two-phase CNP, including subcontracting, to limit subtask negotiations to a subset of initial bidders. Rather than allowing all potential contractors for the high-level task to announce subtasks, the MACRO CNP allows mission agents to announce subtasks only after receiving a pre-accept from the user agent. Further, by limiting the number of pre-accepts a user agent can issue for a given task announcement, the MACRO CNP significantly reduces the total amount of communication and computation overhead in the contract net. One goal of the experiments in Section IV is to determine appropriate cutoffs for the number of pre-accepts allowed under different system configurations and operating conditions. Further, these experiments identify scalability trends for the MACRO pre-commitment subcontracting in terms of the major stress factors related to mission agent capability overlap and task composition.

IV. Subtask Allocation Experiments

This section presents the design and results of experiments that evaluate the performance and scalability of the MACRO extended contract net protocol for subtask allocation under a variety of different potential system configurations and operating conditions. This experimental study of performance with randomly-generated tasks allows us to determine realistic scalability trends and determine appropriate pre-accept cutoffs for real-world applications. These experiments validate our claims in Section III that the MACRO extended contract net, with appropriate pre-accept cutoffs, provides an efficient, scalable solution to the challenges of allocating hierarchically-decomposable tasks in a sensor web or similar large multi-agent system. We determine reasonable limits on the number of pre-accepts required to find the best final bid under a variety of potential system configurations and conditions. Further, we identify scalability trends for the major factors affecting subtask allocation: 1) mission agent capability overlap (defined as a density equal to the average number of mission agents capable of performing requested subtasks), 2) number of alternative decompositions per task, and 3) number of subtasks per task decomposition.

A. Experimental Design

To maintain the generality of our results and their applicability to other large-scale multi-agent systems, we employed a simplified representation of subtasks and mission agent capabilities. Specifically, each subtask is randomly generated in a generic XY plane, and each mission agent is capable of achieving subtasks in a square region within that plane. In this setup, the overlap in capabilities across agents is defined by their overlapping geographic regions. However, this simple representation allows our results to be easily extended to other applications where capabilities are significantly more complex and even unrelated to geography, as long as some estimate of capability overlap or density (*i.e.*, the average number of agents capable of requested subtasks in the system) can be determined. In this experiment, 200 mission agents were grouped into regions of 4 agents with overlapping capabilities. Each group of 4 mission agents was assigned to a tier 2 broker for a total of 50 tier 2 brokers.

We also use a simplified representation of bid characteristics, rather than explicitly model the large range of utility functions a user agent could apply to bids. The experiments employ a single, randomly generated, quality value for each subtask based on the mission agent contracted or subcontracted to perform the subtask. In general, mission agents may be unaware of how the user agent determines bid utility from individual bid characteristics. Therefore a single quality value for each subtask suffices to represent the combined characteristics for that subtask in this experiment. Because mission agents can include multiple possible task decompositions and subcontractors in their bids, the user agent can, in general, choose the one with the highest utility in any given bid.

In each trial, a single user agent and tier 1 broker agent were used to randomly generate a task and its decomposition into subtasks. With the MACRO brokers' ability to aggregate task trees, any task can be decomposed into one or more sets of subtasks, where each subtask can be achieved by at least one mission agent. Therefore, each task in this experiment had a corresponding set of randomly generated decompositions with a variable number of subtasks.

To determine pre-accept limits applicable to a variety of systems, we take a plausibly worst case approach in setting static experimental parameters. For example, in a real system, the different possible decompositions of a task would likely include some of the same subtasks or, at least, many subtasks that could be performed by the same agents. Instead, this experiment considers a worst case scenario where decompositions are completely uncorrelated by independently generating random subtasks for each decomposition. Similarly, subtasks are generated randomly over the range of mission agent capabilities, while a real-world system would likely exhibit greater clustering of subtasks based on geographic location and sensor types. The quality values for subtask bids are generated in the uniform, random range of 1 to 100, which is far more variability in quality than is likely in most systems. Also, all agents capable of performing a subtask bid on the subtask. This results in more messages than in most systems where agents may not be interested in all subtask announcements or may have already committed the requisite resources to another task.

Further, subtask quality can be aggregated in a variety of ways to yield overall task quality, such as using the minimum or maximum value of subtasks (*e.g.*, when quality is determined by timeliness for a set of parallel subtasks). In these experiments, user agents employ a sum quality aggregation function (qaf) to generate the quality of the task from the subtask qualities. This represents a situation in which the utility of a task bid to the user agent depends on the characteristics of each subtask, rather than being determined by a single, limiting subtask, as with the minimum and maximum aggregation functions. Although they are not presented due to space constraints, our experiments also confirmed that the sum qaf was the more difficult parameter setting, requiring more pre-accepts (and messages) to find the best bid than the minimum and maximum qafs.

In these experiments, we define performance in terms of message overhead rather than time overhead because messages are the major factor in the workload for the system infrastructure (*i.e.*, broker agents mediating the contract net negotiations). Sensor webs and similar applications that deal with many task announcements can face significant problems due to communication and computation overhead when allocating each task requires numerous subtask announcements and bids, as with pre-bid subcontracting. Although the sequential pre-accepts in the proposed algorithm require more time to allocate a task than with the pre-bid approach, this provides a disincentive for user agents to increase system workload by making additional pre-accepts after they have received an acceptable final bid. While the increase in time overhead could still be a issue for some applications, it can be significantly mitigated by a simple modification to the limited pre-accept contracting approach. For these applications, the allowed number of pre-accepts could be performed in parallel rather than sequentially, and the presented performance and scalability results for message overhead remain the same.

B. Experimental Results

Each experimental run involved 2000 trials (*i.e.*, 2000 randomly generated tasks). In each trial, the user agent announced a task through the tier 1 broker. The tier 1 broker generated task decompositions and passed this information to the applicable tier 2 brokers, who forwarded the task announcement and decompositions to the applicable mission agents.

In the baseline pre-bid subcontracting approach, each mission agent receiving the task announcement then announced all subtasks that it could not perform through the tier 1 broker. In response, it received bids from all other applicable mission agents for those subtasks. Each mission agent combined these subtask bids and generated a task bid, which was forwarded through its tier 2 broker and the tier 1 broker to the user agent. The user agent ranked the complete bids and chose the one with the highest aggregated quality value. Because the bids were complete, this agent was pre-accepted and contracted. These results were compared to the results with MACRO pre-commitment subcontracting.

In the pre-commitment subcontracting approach, each mission agent receiving the task announcement made a preliminary bid, including only the quality values for the subtasks it could perform. The user agent ranked the preliminary bids by aggregated quality value. Starting with the highest preliminary bid, it pre-accepted the corresponding mission agent. The pre-accepted mission agent then announced all subtasks that it could not perform through the tier 1 broker. In response, it received bids from all other applicable mission agents for those subtasks. It combined the subtask bids and generated a task bid, which was forwarded through its tier 2 broker and the tier 1 broker to the user agent. The user agent continued to pre-accept each preliminary bid in this manner, in order of decreasing quality, to generate the results presented below.

To illustrate the scalability of the MACRO approach, we compare the average number of messages required to reach the best final bid in MACRO pre-commitment subcontracting with the messages required in the baseline pre-bid subcontracting. These results are presented in Figures 3, 4, and 5. In each figure, the circle data points indicate pre-commitment subcontracting and the square data points are the result of pre-bid subcontracting. The 95% confidence interval is shown for each data point, although it is smaller than the data point marker in many cases.

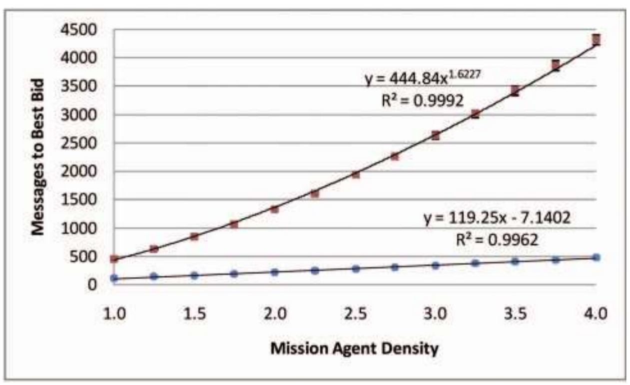

Figure 3. Average messages to best final bid vs. density

The appropriate cutoff for allowed number of pre-accepts depends on the configuration and requirements of the system. For example, the cutoff may be determined based on the desired percentage of tasks that must reach the best final bid. Consequently, this percentage also determines how many messages will be required, on average, in MACRO pre-commitment subcontracting with limited pre-accepts. For a given percentage of trials/tasks to reach the best final bid, the resulting cutoff requires an average number of messages closely correlated with the average number of messages required to reach the best final bid. Therefore, the pre-commitment subcontracting is illustrated by this average number of messages in Figures 3, 4, and 5. Table I provides specific pre-accept cutoff values for at least 75% of trials reaching the best final bid. These results cover a large range of potential system configurations/conditions and illustrate the dramatic reduction of messages in MACRO pre-commitment subcontracting compared to the pre-bid approach.

Figure 3 illustrates the scalability of MACRO pre-commitment subcontracting in terms of mission agent density/overlap. The average number of messages scales linearly with mission agent density. These results were generated with 2-6 (average 4) decompositions per task and 2-6 (average 4) subtasks per decomposition. In comparison, the pre-bid subcontracting approach required an average number of messages that increased with mission agent density to a power of 1.6, for the same parameter settings. For the highest mission agent density of 4.0, pre-bid subcontracting required nearly 10 times the messages of pre-commitment subcontracting.

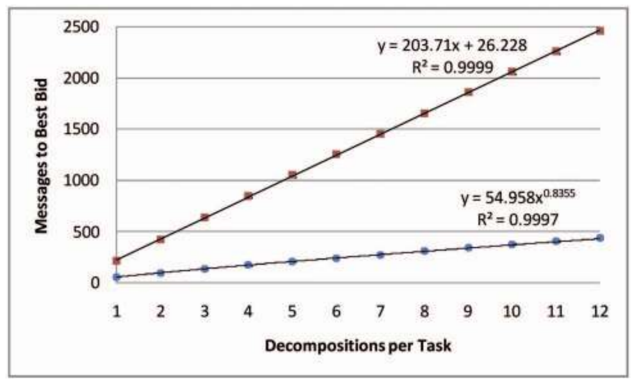

Figure 4. Avg messages to best final bid vs. decompositions

Figure 4 illustrates the scalability of MACRO pre-commitment subcontracting in terms of decompositions per task. The average number of messages scales slightly less than linearly (to the power of 0.8) with the number of decompositions per task. These results were generated with a mission agent density of 1.5 and 2-6 (average 4) subtasks per decomposition. The pre-bid subcontracting approach, on the other hand, scaled linearly in terms of decompositions per task. For the highest decompositions of 12, pre-bid subcontracting required over 5 times the messages of pre-commitment subcontracting.

Figure 5 illustrates the scalability of MACRO pre-commitment subcontracting in terms of subtasks per task decomposition. The average number of messages scales slightly worse than linearly (to the power of 1.2) with number of subtasks. These results were generated with a mission agent density of 1.5 and 2-6 (average 4) decompositions per task. In comparison, the pre-bid subcontracting approach required an average number of messages that increased with mission agent density to a power of 2.0. For the highest subtasks of 12, pre-bid subcontracting required over 10 times the messages of pre-commitment subcontracting.

Density	Decompositions	Subtasks	Pre-Bid Messages	To 75% Cutoff Pre-Accepts	Messages
1.50	1-3	1-3	113 (84)	1	41 (20)
1.50	1-3	4-6	606 (326)	2	140 (41)
1.50	1-3	7-9	1505 (728)	2	226 (58)
1.50	4-6	1-3	276 (117)	3	109 (29)
1.50	4-6	4-6	1494 (408)	3	277 (47)
1.50	4-6	7-9	3714 (840)	4	520 (81)
2.00	1-3	1-3	168 (132)	1	51 (25)
2.00	1-3	4-6	956 (523)	2	172 (51)
2.00	1-3	7-9	2411 (1197)	2	282 (76)
2.00	4-6	1-3	414 (187)	3	136 (38)
2.00	4-6	4-6	2369 (673)	4	400 (72)
2.00	4-6	7-9	5988 (1381)	5	756 (129)

Table I
SUBTASK ALLOCATION RESULTS (2000 TRIALS EACH)

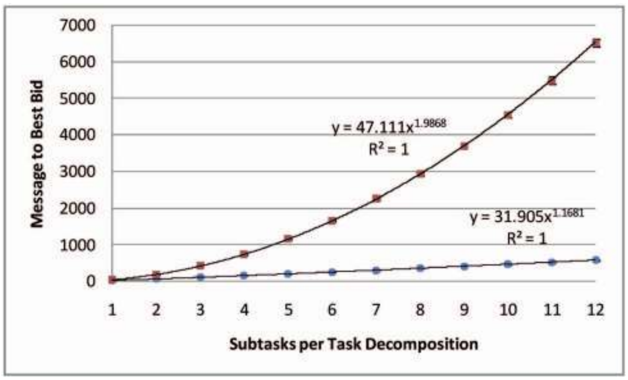

Figure 5. Average messages to best final bid vs. subtasks

V. DISCUSSION AND RELATED WORK

The results presented in Section IV show that the MACRO limited pre-commitment subcontracting approach scales significantly better than the pre-bid subcontracting approach, and that the number of subtasks is the largest stress factor in terms of scalability. Further, the required number of messages to reach the best final bid in pre-commitment subcontracting are on the order of five times fewer than pre-bid subcontracting for likely sensor web system configurations and operating conditions. Considering that each trial illustrates subcontracting of a single task, the number of messages required for pre-commitment subcontracting (*e.g.* 50 to 200 for likely scenarios) are still higher than would be desired. This point is mitigated by the fact that these numbers were determined with plausibly worst case settings of static experimental parameters (*e.g.*, uncorrelated subtasks and task decompositions, all capable agents bidding on task/subtask announcements, and a large range of subtask quality with the application of "sum" quality aggregation function) and by the increasingly low latency and high bandwidth connections available to agents communicating over the internet.

Despite these mitigating factors, the large number of tasks likely to be announced in a global sensor web suggests that further reduction in subcontracting message overhead would be worthwhile in sensor web task allocation. In future work, we intend to include dynamic adjustment of pre-accept cutoff values based on announced task and current configuration/conditions, as well as caching of subtask bids by broker agents, to decrease the number of messages required. Further, we will explore allowing user agents to include information on the utility of specific bid quality criteria to yield smaller, more directed, mission agent bids. These extensions should further diminish the message overhead required in pre-commitment subcontracting, while maintaining the scalability of this approach.

The MACRO approach to allocation of hierarchically-decomposable tasks builds upon, and goes beyond, a significant body of related work. The general problem of "task allocation" has been dealt with in MASs through a variety of techniques suitable to a range of applications, including the Contract Net Protocol (CNP) [6]. A number of extensions to the classic CNP have been proposed to increase performance and handle additional problems presented by various domains. One difficulty in using the classic CNP for sensor web task allocation is that subtasks are interdependent. Sandholm recognized the problem of dependencies between tasks and suggested grouping of tasks in announcements and bidding [8]. However, this solution is insufficient for a global sensor web, in which potential task decompositions are unknown to the user agents announcing tasks. A related approach is to organize agents interested in dependent subtasks into teams. For example, Sims et al. apply bottom-up formation of teams based on marginal utilities to achieve efficient resource coverage in bidding on interdependent tasks/subtasks [9]. However, this approach assumes that many resources are equivalent for a given task, which is often not the case in a global sensor web.

Another significant difference in some variations of the CNP is the determination of when contractors commit to a bid [10]. One solution is to allow bidding on multiple tasks requiring the same resource, plus a conflict resolution stage [11]. Another, more popular solution that can help alleviate the problem of when to bid, is to allow decommitment from contracts [12]–[14]. One approach that is suitable for a variety of applications is to separate initial bidding from commitment (*e.g.*, [7], [15], [16], either with or without decommitments. MACRO employs this approach by extending the two-phase CNP suggested by Aknine et al. [7]. In MACRO, this two-phase CNP is extended with subcontracting during the commitment phase and the use of broker agents to limit the allowed number of pre-accepts. Further, MACRO brokers are employed to aggregate domain information across mission agents and perform preliminary task decomposition into a set of subtask alternatives.

VI. Concluding Remarks

In a multi-agent sensor web system, large-scale tasks may require the resources and capabilities of multiple agents. Employing hierarchically-decomposable tasks allows system designers to deal with the complexity of large tasks and subtasks, but introduces additional challenges for efficient allocation, particularly when individual subtasks could be performed by multiple agents. MACRO employs a novel approach to subtask allocation in a brokered, two-phase contract net. The presented results of experiments using the MACRO limited pre-commitment approach verify its efficiency in terms of scalability and significant reduction in messages over the baseline pre-bid subcontracting approach. Further, these results identify a reasonable upper bound on pre-accept cutoffs for a variety of system configurations and operating conditions.

The generality of representation in these experiments ensures that these results can be extended to other systems in which hierarchically-decomposable tasks must be allocated to agents with overlapping capabilities. In future work, we will investigate dynamic pre-accept cutoffs and caching of subtask bids by broker agents to further reduce message overhead. We will also explore the use of bid preference information to reduce final task bid size in MACRO pre-commitment subcontracting.

References

[1] NASA, "Report from the Earth Science Technology Office (ESTO) Advanced Information Systems Technology (AIST) Sensor Web Technology Meeting," The Aerospace Corporation, Tech. Rep. ATR-2007(2105)-1, May 2007.

[2] D. Suri, A. Howell, D. C. Schmidt, G. Biswas, J. Kinnebrew, W. Otte, and N. Shankaran, "A Multi-agent Architecture for Smart Sensing in the NASA Sensor Web," in *Proceedings of the 2007 IEEE Aerospace Conference*, Big Sky, Montana, Mar. 2007.

[3] J. S. Kinnebrew, "Global Sensor Web Coordination and Control in a Multi-agent System," in *to appear in Proceedings of the AAAI Doctoral Consortium (at IJCAI '09)*, July 2009.

[4] B. Horling, V. Lesser, R. Vincent, T. Wagner, A. Raja, S. Zhang, K. Decker, and A. Garvey, "The TAEMS White Paper," *Multi-Agent Systems Lab, University of Massachusetts*, 1999.

[5] M. Botts *et al.*, "Sensor Model Language (SensorML)," Open Geospatial Consortium, Tech. Rep. OpenGIS Implementation Specification Document 07-000, July 2007. [Online]. Available: http://www.opengeospatial.org/standards/sensorml

[6] R. Smith, "The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver," *IEEE Transactions on Computers*, vol. 29, no. 12, pp. 1104–1113, 1980.

[7] S. Aknine, S. Pinson, and M. Shakun, "An Extended Multi-Agent Negotiation Protocol," *Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 1, pp. 5–45, 2004.

[8] T. Sandholm, "An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations," *Proceedings of the National Conference on Artificial Intelligence*, pp. 256–256, 1993.

[9] M. Sims, C. Goldman, and V. Lesser, "Self-organization through Bottom-up Coalition Formation," *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pp. 867–874, 2003.

[10] M. Schillo, C. Kray, and K. Fischer, "The Eager Bidder Problem: A Fundamental Problem of DAI and Selected Solutions," *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pp. 599–606, 2002.

[11] D. Ouelhadj, C. Hanachi, B. Bouzouia, A. Moualek, and A. Farhi, "A Multi-Contract Net Protocol for Dynamic Scheduling in Flexible Manufacturing Systems (FMS)," *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, 1999.

[12] T. Sandholm and V. Lesser, "Issues in Automated Negotiation and Electronic Commerce: Extending the Contract Net Protocol," *Proceedings of the First International Conference on Multiagent Systems (ICMAS-95)*, 1995.

[13] J. Vokrınek, J. Bıba, J. Hodık, J. Vybıhal, and M. Pechoucek, "Competitive Contract Net Protocol," *Proceedings of the 33rd Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, pp. 656–668, 2007.

[14] E. Tsang, T. Gosling, B. Virginas, C. Voudouris, G. Owusu, and W. Liu, "Retractable Contract Network for Empowerment in Workforce Scheduling," *Multiagent and Grid Systems*, vol. 4, no. 1, pp. 25–44, 2008.

[15] T. Knabe, M. Schillo, and K. Fischer, "Improvements to the fipa contract net protocol for performance increase and cascading applications," in *Proceedings of the International Workshop for Multi-Agent Interoperability at the Annual German Conference on AI (KI-2002)*, 2002.

[16] D. Weyns, N. Boucke, T. Holvoet, and K. Schelfthout, "DynCNET: A Protocol for Flexible Task Assignment Applied in an AGV Transportation System," *Proceedings of the Fourth European Workshop on Multi-Agent Systems*, pp. 359–370, 2006.